**CS 260 Self Evaluation for Assignment 6 – Recursive Sorts**

|  |  |
| --- | --- |
| Your name:  David Oftedahl | Date:  12/04/2022 |
| Are you willing to allow your code to be used in example debugging demonstrations or documentation?  Yes  No | |

**Instructions – Part 1**  
This document is to be turned in alongside solution of this lab. You will use this document to indicate your status on the lab, as well as areas where you are struggling conceptually or in converting concept to code. Please use the space underneath each evaluation criteria to describe any errors you are receiving or challenges you are having implementing the required functionality for your code.

**Project Organization (please fill in for your language)**

|  |  |
| --- | --- |
| ***File Structure (C++)*** | |
| Is your class split into a header and a source file? |  |
|  | |
|  | |
| ***File Structure (C#)*** | |
| Is your class declared in a class library, separate from your console driver? |  |
|  | |
|  | |
| ***File Structure (Python)*** | |
| Is your class declared in a separate, imported file? |  |
|  | |

**Functionality**

|  |  |
| --- | --- |
| **Base Lab** | |
|  | |
| ***Heap*** | |
| Does the program compile without errors or warnings and run without crashing? |  |
|  | |
| Does the class properly add and remove items? |  |
|  | |
| Does it meet the test examples properly? |  |
|  | |
| ***Priority Queue*** | |
| Does the program compile without errors or warnings and run without crashing? |  |
|  | |
| Is it built as a wrapper (composition) upon a heap instance? |  |
|  | |
| Does the class properly add and remove items? |  |
|  | |
| **Advanced Lab** | |
|  | |
| ***HeapSort*** | |
| Does the program compile and run without errors or warnings and run without crashing? |  |
|  | |
| Does it properly sort the array from smallest to largest? |  |
|  | |
| Does it work by heapifying the data and then taking it away one item at a time? |  |
|  | |

|  |  |
| --- | --- |
| ***MergeSort*** | |
| Does the program compile without errors or warnings and run without crashing? |  |
|  | |
| Does it properly sort the array from smallest to largest? |  |
|  | |
| Does it work by recursively breaking the array into halves and then merging them? |  |
|  | |
| ***QuickSort*** | |
| Does the program compile without errors or warnings and run without crashing? |  |
|  | |
| Does it properly sort the array from smallest to largest? |  |
|  | |
| Does it work by selecting a pivot and sorting partitioned values relative the pivot value? |  |
|  | |
| **Thinking Problem** | |
|  | |
| ***FindNth*** | |
| Does the program compile without errors or warnings and run without crashing? |  |
|  | |
| Does it properly stop sorting once the nth value has been found? |  |
|  | |

**Instructions – Part 2**   
Please answer the following questions, in your own words, regarding your experiences throughout this lab.

**Experiential Review**

|  |
| --- |
| **What aspects of this lab did you find most challenging?** |
|  |
| **What concept from this lab do you feel you have the best grasp on now?** |
|  |
| **Please summarize the basic concepts of how each of these sorts functions and what the pros and cons are for each:** |
|  |
| **When might you choose to use an insertion or selection sort instead of one of the recursive sorts covered this week?** |
|  |